Abstract—Blind source separation is the problem of extracting independent signals from their mixtures without knowing the mixing coefficients nor the probability distributions of source signals and may be applied to EEG and MEG imaging of the brain. It is already known that certain algorithms work well for the extraction of independent components. The present paper is concerned with superefficiency of these based on the statistical and dynamical analysis. In a statistical estimation using $t$ examples, the covariance of any two extracted independent signals converges to 0 of the order of $1/t$. On-line dynamics shows that the covariance is of the order of $\eta$ when the learning rate $\eta$ is fixed to a small constant.

In contrast with the above general properties, a surprising superefficiency holds in blind source separation under certain conditions where superefficiency implies that covariance decreases in the order of $1/t^2$ or of $\eta^2$. The present paper uses the natural gradient learning algorithm and method of estimating functions to obtain superefficient procedures for both batch estimation and on-line learning. A standardized estimating function is introduced to this end. Superefficiency does not imply that the error variances of the extracted signals decrease in the order of $1/t^2$ or $\eta^2$ but implies that their covariances (and independencies) do.

Index Terms—Blind source separation, error analysis, estimating function, independent component analysis, on-line learning, superefficiency.

I. INTRODUCTION

Blind source separation is the problem of extracting independent signals when their mixtures are observed. Herault and Jutten [18] proposed an attractive idea inspired by neural learning [20], and a lot of algorithms have been suggested since then, including the independent component analysis (ICA) [10], [17], maximum likelihood [26], entropy maximization [13], [23], nonlinear PCA [24], and algebraic approach [14]. Yang and Amari [30] elucidated the relation between the ICA and entropy maximization approaches. The statistical efficiency of algorithms is analyzed in [8], [15], [27], and others, whereas a more fundamental treatment is given from the point of view of semiparametric estimation [7]. The natural or relative gradient has also been introduced to guarantee equivariant properties [10], [15]; see also [4]. Dynamic stability of learning has been analyzed under certain conditions in [15], [16], [22], and [28]. Amari et al. [8] have succeeded in providing stability analysis under general conditions and proposed a universally convergent learning algorithm.

Computer simulation studies have also established that learning algorithms sometimes work surprisingly well for the extraction of independent signals. More interestingly, these algorithms have been applied recently to brain imaging data [21], and a number of better-than-expected, independent components have been extracted. The present paper describes it as “superefficient” and uses the statistical and dynamical analysis to explain superefficiency.

When the unknown mixing matrix or its inverse is estimated from $t$ examples, the batch estimation errors measured by the variance-covariance matrix decrease on the order of $1/t$ when $t$ is large, thus implying that the covariance of any two extracted signals similarly decreases. Under certain conditions, however, there emerges a surprising superefficiency of independent source extraction in which the covariance of two signals decreases as $1/t^2$ rather than $1/t$. A statistical description of these conditions under which superefficiency occurs is offered in the present paper.

Superefficiency of on-line learning is then discussed. When the learning rate $\eta$ is fixed to a small constant, the dynamical theory of on-line learning [1], [19] proves that the covariance of extracted signals is, in general, of order $\eta$ for large $t$. This shows that fluctuations of order $\eta$ remain after learning, whereas the convergence speed of learning depends solely on $\eta$. Here, we prove that superefficiency emerges even in on-line learning. Superefficiency in this case implies covariance of the order of $\eta^2$ for large $t$. Superefficiency of on-line learning is proved under the same conditions.

The paper uses the method of estimating functions [12] to analyze the present problem of semiparametric statistical models. Among the class of equivalent estimating functions, a standardized estimating function is defined and explicitly calculated, and the standardized estimating function is proved to give the best on-line performance. This result is used to prove superefficiency both in batch and on-line estimation procedures.

The asymptotic equivalence of batch and on-line learning procedures is confirmed finally by extending Amari [3], [4] and Opper [25]. Superefficiency is also expected to apply to the problem of multichannel blind deconvolution in which natural gradient is studied by Amari et al. [11].

II. STATEMENTS OF THE PROBLEM AND MAIN RESULTS

Let us consider $n$ independent sources of signals generating $n$ signals $s_a(t), a = 1, \ldots, n$ at discrete times $t = 1, 2, \ldots$. They may be represented by the column vector

$$s(t) = [s_1(t), s_2(t), \ldots, s_n(t)]^T$$
where $T$ denotes the transposition. It is assumed that $s_a(t)$ are normalized white random variables with mean 0, that is

\[
E[s_a(t)] = 0
\]
\[
E[s_a(t)s_a(t')] = \delta_{ab}\delta_{tt'}
\]

where $E$ denotes expectation, and $\delta$ denotes the Kronecker delta.

An assumption is made that the signals $s(t)$ cannot be directly observed but that their instantaneous mixtures $x(t) = [x_1(t), \cdots, x_n(t)]^T$ are observed. Here, $x(t)$ is represented as

\[
x(t) = As(t)
\]

or in component form as

\[
x_i(t) = \sum_{a=1}^{n} A_{ai}s_a(t).
\]

The present paper assumes that the number of source and observed signals is the same (see [6] in the situations in which they may be different). The mixing matrix $A$ is nonsingular and unknown. Blind source separation is the recovery of the original signals $s(1), s(2), \cdots$ from observed signals $x(1), x(2), \cdots$. The description “blind” implies that we do not know the mixing matrix $A$ nor the probability distributions of $s_a$’s except that they are independent.

If we know $W = A^{-1}$, the original signals are easily recovered by

\[
y(t) = Wx(t)
\]

\[
y_a(t) = \sum_{i=1}^{n} W_{ai}x_i(t).
\]

Therefore, the problem is to estimate $W = (W_{ai})$ by using $x(1), \cdots, x(t)$ from a statistical point of view and then obtain an estimate $y(t + 1)$ of $s(t + 1)$ by

\[
y(t + 1) = W_{t}x(t + 1)
\]

where $W_t$ is an estimate of $W$ from $t$ observations $x(1), \cdots, x(t)$. The estimator $W_t$ is derived either by statistical batch procedure or recursively by on-line learning. On-line learning is carried out by a dynamical equation of the type

\[
W_{t+1} = W_t - \eta F[x(t), W_t]
\]

where $\eta$ is a learning constant and $F$ a suitable matrix function of $x$ and $W$. It should be noted that the mixing matrix $A$ and its inverse $W$ are not identifiable. Even if we can extract $n$ independent signals, we do not know how to arrange them in order. Hence, the recovered signals $y_1, \cdots, y_n$ can be arranged in any permutation of the original $s_1, \cdots, s_n$. Moreover, the absolute scale of each $s_a$ is not identifiable. This is because multiplying $s_a$ by a scalar $c_a$ is equivalent to multiplying the $a$th column of $A$ by the same $c_a$. Therefore, the original independent components $s_a(t)$ are recovered, except for their scales and ordering. In order to reduce the scaling indefiniteness, we usually introduce the normalization constraint

\[
E[h_a(s_a)] = 0
\]

where $h_a$ is an arbitrary function, for example

\[
h_a(s) = s^2 - 1.
\]

Then, $W$ can be determined uniquely except for permutation. Recently, nonholomic constraints have been introduced in [9], and these work much better when the original sources are nonstationary.

Let

\[
y(t + 1) = W_{t}x(t + 1)
\]

be a set of recovered signals. The main topic of the present paper is to see how independent the two extracted signals $y_a(t)$ and $y_b(t)$, $a \neq b$ are via statistical estimation or, in particular, by on-line learning. To this end, by noting that $E[y(t)] = 0$, we put

\[
V_t = E[y(t)y^T(t)]
\]

or in component form

\[
V_{\theta t} = E[y_a(t)y_b(t)].
\]

It is proved from standard asymptotic statistical theory and on-line learning [1] that $V_{\theta t}$ decreases of the order of $1/t$ under batch estimation and of the order of the learning constant $\eta$ under on-line learning.

Our main results are to find superefficiency implying that $V_{\theta t}$ converges much faster when a certain condition is satisfied. In order to state this condition, we define a matrix-valued function

\[ F_{\varphi}(y, W) = \varphi(y)y^T - I \]

where $I$ is the $n$ by $n$ identity matrix, and $\varphi(y)$ is a vector composed of

\[ \varphi(y) = [\varphi_1(y_1), \cdots, \varphi_n(y_n)]^T \]

with arbitrary differentiable nonlinear functions $\varphi_i$. In order to obtain a batch estimator $W_t$, we use the solution of the estimating equation

\[
\sum_{i=1}^{t} F_{\varphi}(y(i), W_t) = 0
\]

where $y(i) = W_t x(i)$. In order to obtain an on-line estimator, we use the natural gradient learning algorithm [4], [8]

\[
W_{t+1} = W_t - \eta F_{\varphi}(y(t), W_t)W_t.
\]

Now, the main result is stated as follows.

**Main Result: Superefficiency, in the sense that covariance $V_{\theta t}$ decreases of the order of $1/t^2$ in batch estimation and of the order of $\eta^2$ in on-line learning, holds when condition $A$ is satisfied:**

\[ E[\varphi_a(s_a)] = 0. \]

It should be noted that Condition $A$ holds in the following two cases.
Case 1: The condition is satisfied when

$$
\phi_a(s_a) = -\frac{d}{ds_a} \log r_a(s_a) 
$$

(2.16)

where $r_a(s_a)$ are the true probability density functions of the source signals $s_a$. In this case, $W_t$ is the maximum likelihood estimator. In blind source separation, the functions $r_a(s_a)$ are usually unknown. Therefore, it is difficult to choose $\phi_a$ such that it satisfies (2.16).

Case 2: The functions $r_a(s_a)$ are even, and $\phi_a(s_a)$ are odd functions. In many applications, $r_a$ are even; therefore, Condition $A$ is satisfied when odd $\phi_a(s_a)$ are chosen, as is done in most cases of applications. This is why super-efficiency holds in various applications.

This condition is used also in a paper [27] that calculates the asymptotic covariances of estimators of $A$. Super-efficiency result of batch estimation may be derived from their approach, but on-line learning is different.

III. ESTIMATING FUNCTIONS

The problem of blind source separation is formulated in the framework of a semiparametric statistical model [7], [12]. Let $r_a(s_a)$ be the probability density function of $s_a$. The joint probability density of $s$ is written as

$$
r(s) = \prod_{a=1}^{n} r_a(s_a)
$$

(3.1)

since they are independent. The observation vector $x$ is a function of $s$, therefore, its probability density function is given in terms of $W = A^{-1}$ by

$$
p_x(x; W, r) = \det |W| r(W x).
$$

(3.2)

Since we do not know $r$ except that it is a product form (3.1), the probability model of $x$ includes two parameters: $W$, which is known as the “parameter of interest” (which we want to estimate) and the “nuisance parameter (function)” $r = r_1 \cdots r_n$, which is of no consequence. Such a statistical model is called a semiparametric model, and estimation of the parameter of interest is, in general, a difficult problem because of the existence of unknown functions.

A method of estimating functions has been developed for semiparametric statistical models, and its mathematical foundation is supplied by information geometry [12]; see [2] and [5] for information geometry.

An estimating function in the present case is a matrix-valued function $F(x, W) = \{F_{ab}(x, W)\}$ of $x$ and $W$, not including the nuisance parameter $r$, that satisfies

1) $E_{W,r}[F(x, W)] = 0.
$$

(3.3)

Here, $E_{W,r}$ denotes expectation with respect to probability distribution given by (3.2), and it is required that (3.3) holds for all $r$ of the form (3.1). In order to avoid a trivial $F$ such as $F = 0$, we require that

2) $H = E_{W,r} \left[ \frac{\partial}{\partial W} F(x, W) \right]
$$

(3.4)

is non-degenerate. It should be noted that $H$ is a matrix-by-matrix operator that maps matrix to matrix in a linear fashion. The components of $H$ are

$$
H_{ab, ci} = E_{W,r} \left[ \frac{\partial}{\partial W_{ci}} F_{ab}(x, W) \right].
$$

It is convenient to use capital indices $A, B, \cdots$ to represent a pair $(a, b), (c, i)$, and so on, of indices. Then, for $A = (a, b), B = (c, i)$, $H$ has a matrix representation $H = (H_{AB})$ that operates on $(W_B) = (W_{ci})$ as

$$
HW = \sum_{B} H_{AB} W_B = \sum_{c,i} H_{ab, ci} W_{ci}.
$$

The inverse of $H$ is defined by the inverse matrix of $H = (H_{AB})$.

When an estimating function $F(x, W)$ is found, we have the estimating equation

$$
\sum_{i=1}^{t} F\{x(i), W\} = 0
$$

(3.5)

the solution to which provides estimator $W_t$. This is derived by replacing the expectation in (3.3) by the empirical sum of observations. The equation is solved without making use of the unknown $r$. The problem is how to find a “good” estimating function $F$.

A number of heuristic estimating functions have been proposed including [10], [13], [20], and [24]. While mathematical theory proves [7] that estimating functions of the form

$$
F(x, W) = \phi(y) y^T - I
$$

(3.6)

or

$$
F_{ab}(x, W) = \phi_a(y_a) y_b - \delta_{ab}
$$

in component form

$$
\phi(y) = [\phi_1(y_1), \cdots, \phi_n(y_n)]^T
$$

for arbitrary nontrivial functions $\phi_a$ span all the effective estimating functions. This implies that given any estimating function, an equivalent or better estimating function can be found in the class spanned by (3.6). Moreover, this class includes the best estimator (that is, the Fisher efficient estimator) in the sense that it satisfies the extended Cramér–Rao bound asymptotically.

When the true distributions are $r_a$, the best choice of $\phi_a$ is

$$
\phi_a(s) = -\frac{d}{ds_a} \log r_a(s).
$$

(3.7)

This gives the maximum likelihood estimator [13], [26]. However, even when we use a different $\phi_a$, the estimating equation (3.5) gives a $\sqrt{t}$-consistent estimator, that is, the estimation error converges to 0 in probability of the order of $1/\sqrt{t}$ as $t$ goes to infinity.

In the case of on-line learning, an estimating function $F$ gives a learning algorithm

$$
W_{t+1} = W_t - \eta F\{x(t), W_t\}.
$$

(3.8)

However, an important difference between batch estimation and on-line learning should be noted. To demonstrate this, we
introduce an equivalence relation in the class of estimating functions.

Let $R(W)$ be an arbitrary nonsingular linear operator acting on matrices. When $F(x; W)$ is an estimating function matrix, $R(W)F(x; W)$ is also an estimating function matrix because

$$E_{W_t}[R(W)F(x; W)] = R(W)E_{W_t}[F(x; W)] = 0.$$  

Moreover, $F$ and $RF$ are equivalent in the sense that the derived batch estimators are exactly the same because the two estimating equations

$$\sum_i F(x_i; W) = 0$$

$$\sum_i R(W)F(x_i; W) = 0$$

give the same solution $W_t$. This defines an equivalent class of estimating functions that are essentially the same in batch estimation.

However, two equivalent estimating functions $F(x; W)$ and $R(W)F(x; W)$ give different dynamical properties in on-line learning. Therefore, instead of the form (3.6), we need to consider an enlarged type of estimating functions of the form $R(W)F$ to derive a good on-line estimator. However, $F$ and $RF$ are not equivalent with regard to on-line learning. The standardized estimating function will be introduced in Section V to this end. See also [8].

### IV. General Error Analysis

We now give the standard statistical error analysis. See also [27]. The estimation error is given by

$$\Delta W_t = W_t - W.$$  

Let us define the relative error matrix $\Delta X_t = (\Delta X_{e_i})$ by

$$\Delta X_t = \Delta W_t W_t^{-1}.  \tag{4.1}$$

This quantity is convenient for our analysis because the recovering error is written as

$$\Delta s = \Delta W_t x = \Delta X_t s.  \tag{4.2}$$

We now calculate the estimation error $\Delta X_t$. Via asymptotic statistical analysis, we expand the estimating (3.5) to

$$0 = \sum_{i=1}^{t} \frac{\partial F(x_i; W)}{\partial W} \Delta X_t W$$

$$= \sum_i F(x_i; W) + \sum_i \frac{\partial F(x_i; W)}{\partial W} \Delta X_t W.  \tag{4.3}$$

We use the notation

$$\partial F/W = \partial F/\partial W$$

in component form

$$\frac{\partial F_{ab}}{\partial X_{cd}} = \sum_i \frac{\partial F_{ab}}{\partial W_{cd}} W_{di}.$$  

From (4.3), we derive

$$\frac{1}{\sqrt{t}} \sum_{i=1}^{t} \frac{\partial F(x_i; W)}{\partial X} \Delta X_t = -\frac{1}{\sqrt{t}} \sum_{i=1}^{t} F(x_i; W).  \tag{4.4}$$

When $t$ is large, the law of large numbers guarantees

$$\frac{1}{t} \sum_{i=1}^{t} \frac{\partial F(x_i; W)}{\partial X} \Delta X_t = K + O_p\left(\frac{1}{\sqrt{t}}\right)  \tag{4.5}$$

where

$$K = E \left[ \frac{\partial F(x; W)}{\partial X} \right]  \tag{4.6}$$

and $K = (K_{ab,cd})$ is a matrix-to-matrix operator. Since the expectation of $F(x; W)$ itself is zero, the central limit theorem guarantees that the right-hand side of (4.4)

$$\frac{1}{\sqrt{t}} \sum F(x_i; W)$$

converges in distribution to the normal random variable matrix with mean 0 and covariances given by $G$

$$G_{ab,cd} = E[F_{ab}(x; W)F_{cd}(x; W)].$$

From this, we have the following lemma.

**Lemma 1:** The covariance of the error measured in terms of $\Delta X_t$ of the estimator $W_t$ is asymptotically given by

$$E[\Delta X_t \Delta X_t^T] = \frac{1}{t} K^{-1} G K^{-1} + O\left(\frac{1}{t^2}\right)  \tag{4.7}$$

where $K^{-1}$ is the inverse of the operator $K$, or by putting $A = (a, b), B = (c, d)$

$$E[\Delta X_A \Delta X_B^T] = \frac{1}{t} \sum_{C,D} K_{AC}^{-1} G_{CD} K_{DB}^{-1} + O\left(\frac{1}{t^2}\right)$$

where $G_{CD}$ is the abbreviated form of $G = (G_{ab,cd})$, and $K_{AC}^{-1}$, $K_{BC}^{-1}$ are the components of the inverse of $K$.

**Proof:** By substituting (4.5) for the left-hand side of (4.4), we have

$$\Delta X_t = -\frac{1}{\sqrt{t}} K^{-1} \left\{ \frac{1}{\sqrt{t}} \sum_{i=1}^{t} F(x_i; W) \right\}.  \tag{4.8}$$

Hence, $\Delta X_t$ is asymptotically normally distributed with mean 0. Its covariance is given by (4.7).

The covariance matrix $V_t$ of the recovered signals is easily calculated from the covariances of the estimator $W_t$ or $\Delta X_t$.

**Lemma 2:** For $a \neq b$

$$V_{ab} = E[\Delta s_a \Delta s_b] = E[y_a(t)y_b(t)] = \sum_c E[\Delta X_{ac}\Delta X_{bc}^T] \sigma_c^2  \tag{4.9}$$

where

$$\sigma_c^2 = E[s_c^2]  \tag{4.10}$$

is determined from the constraint (2.7).
Proof: From (4.2), we have, for \( a \neq b \)
\[
E[y_a(t)y_b(t)] = E[s_a(t) + \sum_c \Delta X_{ac}s_c(t)][s_b(t) + \sum_d \Delta X_{bd}s_d(t)]
\]
\[
= E[\Delta s_a \Delta s_b] + \sum_{c,d} E[\Delta X_{ac}\Delta X_{bd}s_c s_d]
\]
\[
= \sum_{c,d} E[\Delta X_{ac}\Delta X_{bd}]E[s_c s_d]
\]
\[
= \sum_c E[\Delta X_{ac}\Delta X_{bc}]\sigma^2_c.
\]
Here, we used
\[
E[\Delta X_{ac}] = 0
\]
which is proved immediately from (3.3) and (4.8).

V. STANDARDIZED ESTIMATING FUNCTION

For each equivalence class of estimating functions, it is useful to establish a standardized one since this has an essential role in on-line learning.

Estimating function \( F^* \) is said to be standardized when
\[
K^* = E \left[ \frac{\partial}{\partial X} F(x; W) \right]
\]
(5.1)
is the identity operator.

Lemma 3: Given an estimating function \( F \), its standardized form is given by
\[
F^* = K^{-1}(W)F(x; W)
\]
(5.2)
where \( K \) is given by (4.6). When the standardized \( F^* \) is given, we have
\[
E[\Delta X \Delta X] = \frac{1}{t} E[F^* F^*] + O \left( \frac{1}{t^2} \right)
\]
(5.3)
where \( \Delta X \Delta X \) and \( F^* F^* \) are not matrix multiplication but the direct product implying \( \Delta X_{ab} \Delta X_{cd} \) and \( F^*_{ab} F^*_{cd} \), respectively, in component form.

Proof: For \( R(W)F(x; W) \), we have
\[
E \left[ \frac{\partial}{\partial X} \left( R(W)F(x; W) \right) \right]
\]
\[
= \left\{ \frac{\partial}{\partial X} R(W) \right\} E[F(x; W)] + R(W) E \left[ \frac{\partial}{\partial X} F(x; W) \right]
\]
\[
= R(W)K(W).
\]
Hence, when \( R(W) = K^{-1}(W) \), we have
\[
K^* = E \left[ \frac{\partial}{\partial X} F^*(x; W) \right] = K^{-1}K = \text{identity}.
\]

Equation (5.3) is based on (4.7).

Given \( F \) of the form (3.6), in order to obtain its standardized estimating function explicitly, we calculate the operator \( K = (K_{ab, cd}) \)
\[
K_{ab, cd} = E \left[ \frac{\partial F_{ab}(x; W)}{\partial X_{cd}} \right].
\]

When \( F \) is given by the gradient of expectation of scalar cost function, \( K \) is its Hessian. The Hessian was calculated in [8], [15], [26], and [27].

Lemma 4: At the true value \( W \) where \( y_a = s_a \) is recovered, for \( F \) given by (3.6), \( K \) is calculated as
\[
K_{ab, cd} = E[\varphi'(s_a)s_a^2] \delta_{ac} + \delta_{ad}\delta_{bc}
\]
(5.4)
where \( \varphi' \) denotes the derivative of \( \varphi \).

The proof is given in Appendix A. Now, let us put
\[
n_a = E[\varphi'(s_a)]
\]
(5.5)
\[
k_a = E[\varphi'(s_a)]
\]
(5.6)
Then
\[
\frac{\partial F_{ab}}{\partial X_{cd}} = \begin{cases} 1 + n_a, & \text{when } c = a, d = a \\ 0, & \text{otherwise} \end{cases}
\]

We know that for \( a \neq b \)
\[
\frac{\partial F_{ab}}{\partial X_{cd}} = 0
\]
unless \((a, b) = (c, d)\) or \((a, b) = (d, c)\). When the pairs \((a, b)\) and \((c, d)\) are equal, (4.5) gives
\[
K_{ab, cd} = k_a \sigma^2_a
\]
\[
k_{ab, cd} = 1.
\]

Let us summarize the above results. To this end, we denote \( K \) in the pairwise component form of the enlarged matrix \( K = (K_{AB}) \). The results show that for \( A = (a, a) \), \( K_{AB} = 0 \) except for \( B = A \). For \( A = (a, b) \), \( a \neq b \), \( K_{AB} = 0 \) except for \( B = (a, b) \) or \( B = (b, a) \). This shows that \( K = (K_{AB}) \) has a diagonalized part for \( A = (a, a) \) and, for \( A = (a, b) \), \( a \neq b \), its two-by-two submatrix is composed of the two-by-two minor diagonal matrix of \( \partial F_{ab}/\partial X_{ab} \), \( \partial F_{ab}/\partial X_{ba} \), \( \partial F_{ab}/\partial X_{ca} \), and \( \partial F_{ab}/\partial X_{cb} \)
\[
K^{-1}_{AA} = (K_{AA})^{-1}
\]
and for the \((A, A')\) part
\[
\begin{bmatrix} k_a \sigma^2_a & 1 \\ 1 & k_b \sigma^2_b \end{bmatrix}^{-1} = c_{ab}\begin{bmatrix} k_a \sigma^2_a & -1 \\ -1 & k_b \sigma^2_b \end{bmatrix}
\]
(5.8)
where
\[
c_{ab} = \frac{1}{k_a k_b \sigma^2_a \sigma^2_b - 1}.
\]

From the standardized form of the estimating function, (5.2) is calculated as follows.
Theorem 1: The standardized estimating function matrix \( F^* \) is given by
\[
F^*_{aa} = \frac{1}{n_a + 1} \left( \varphi_a(y_a)y_a - 1 \right) \\
F^*_{ab} = c_{ab} \left( \varphi_a(y_a)y_b - \varphi_b(y_b)y_a \right), \quad a \neq b.
\]
(5.10) (5.11)

Proof: This follows directly from (5.2) and (5.8).

VI. SUPEREFFICIENCY OF BATCH ESTIMATORS

The covariance matrix \( V_t \) is now calculated explicitly. To this end, we put
\[
I_a = E[\varphi_a(s_a)] \\
C_{ab, cd} = E[F^*_{ab}(x, W)F^*_{cd}(x, W)].
\]
(6.1) (6.2)

Lemma 5: The covariances of \( \Delta X^*_t \) are given as
\[
E[\Delta X^*_a \Delta X^*_c] = \frac{1}{t} C^*_{ac} + O \left( \frac{1}{t^2} \right).
\]
(6.3)

In particular
\[
C^*_{ac} = \begin{cases} 
\frac{1}{n_a + 1} \alpha_a \sigma^2_c & a \neq b, \\
\frac{1}{n_a + 1} \alpha_a \sigma^2_c & a \neq b, \\
\frac{1}{n_a + 1} \alpha_a \sigma^2_c & a \neq b.
\end{cases}
\]
(6.4) (6.5)

Proof is given in Appendix B.

Lemma 6: The covariance matrix \( V_t \) is given by
\[
V^*_{aa} = \frac{1}{t} \sum_b G^*_{ab, ab} + O \left( \frac{1}{t^2} \right) \\
V^*_{ab} = \frac{1}{t} \sum_c G^*_{ac, c} + O \left( \frac{1}{t^2} \right), \quad (a \neq b).
\]
(6.6) (6.7)

The lemma shows that the expected squared errors \( V^*_a = E[(\Delta s_a)^2] \) decrease in proportion to \( 1/t \), and the covariances \( V^*_a = E[\Delta s_a \Delta s_b] = E[y_a y_b] \) of the recovered signals \( y_a \) and \( y_b \) also decrease in the same order. This fact agrees with ordinary asymptotic statistical analysis. However, we can prove a superefficiency implying dependency of any two recovered signals decreases in the order of \( 1/t^2 \) under Condition A. This indicates that independency is easily found by some algorithms.

Theorem 2: A batch estimator is superefficient when Condition A is satisfied.

Proof: When Condition A holds, we have \( I_a = 0 \); therefore, \( V^*_{ab} (a \neq b) \) satisfies
\[
V^*_{ab} = O \left( \frac{1}{t^2} \right).
\]

The coefficients of \( V^*_{ab} \) of order \( 1/t^2 \) can be calculated explicitly by higher order asymptotics of statistical inference [2]. They include \( c \)- and \( m \)-curvature terms of the statistical manifold as well as the connection terms.

VII. SUPEREFFICIENCY IN ON-LINE LEARNING

We have, thus far, studied the statistical analysis of the asymptotic errors \( \Delta s \) when \( W \) is estimated by a batch type algorithm where all past data are stored. In many cases, blind separation is carried out on-line.

The natural or relative gradient learning rule [10], [15] is given by
\[
W_{t+1} = W_t - \eta F(x_t, W_t)W_t
\]
(7.1)
where \( F \) (and hence \( FW \)) are estimating functions of the form \( F = F(y) \). Here, we first assume
\[
F(y) = \varphi(y)y^T - I.
\]

Let us denote the increment of \( W_t \) as
\[
\delta W_t = W_{t+1} - W_t
\]
and define
\[
\delta X_t = \delta W_t W_t^{-1}.
\]
(7.2)

The learning equation can be written as
\[
\delta X_t = -\eta F(x_t, W_t),
\]
(7.3)

In order to analyze the asymptotic behavior of the learning dynamics, we recapitulate here a previous account of dynamics of on-line learning [1] rediscovered in Heskes and Kappen [19]. When \( \eta \) is a small constant, it is proved that the expectation of \( W_t \) converges to optimal value exponentially, provided the learning algorithm is stable. Even when \( t \) is large, however, \( W_t \) still fluctuates around the optimal value. Its variance-covariance matrix is given by the following theorem. A rough proof is given in Appendix C. We remark here that \( W_t \) might converge to a local minimum. Therefore, our analysis is local.

Theorem 3: When \( \eta \) is sufficiently small and \( W_t \) converges to the true solution, the covariance matrix of the relative error
\[
\Delta X_t = (W_t - W)W^{-1}
\]
converges to
\[
E[\Delta X_A \Delta X_B] = \eta Y_{A,B} + O(\eta^2)
\]
where \( A = (a, b) \), \( B = (c, d) \), and \( Y_{A,B} \) is given by the solution of
\[
\sum_C Y_{A,C} K_{BC} + \sum_C Y_{B,C} K_{AC} = G_{AB}.
\]
(7.4)

We now apply the above theorem to our argument. Here, a learning algorithm is said to be superefficient when \( V^*_{ab} = E[y_a(t)y_b(t)] \), \( a \neq b \) is of the order \( \eta^2 \) for sufficiently large \( t \).

Theorem 4: Superefficiency holds for the natural gradient on-line learning algorithm when Condition A is satisfied.
Proof: We have already shown that $K_{BC}$ is block diagonalized. Therefore, (7.4) splits into blocks. In order to calculate

$$V_{ab} = \eta \sum_c Y_{ac,bc}\sigma_c^2 + O(\eta^2), \quad a \neq b \quad (7.5)$$

we need to calculate $Y_{ac,bc}$ for $a \neq b$. When $c$ is different from $a$ and $b$, (7.4) splits into the closed blocks consisting of four variables $Y_{ac,bc}, Y_{ac,cb}, Y_{ca,bc}$, and $Y_{ca,cb}$. Equation (7.4) for these four variables is

$$\begin{bmatrix}
    k_c\sigma_c^2 & 1 & k_a\sigma_a^2 & 1 \\
    1 & k_c\sigma_c^2 & 1 & k_a\sigma_a^2 \\
    1 & k_c\sigma_c^2 & 1 \quad 1 & k_a\sigma_a^2 \\
    1 & \text{l}_{ac,bc} & \text{l}_{ac,cb} & \text{l}_{ca,bc} & \text{l}_{ca,cb}
\end{bmatrix} = \begin{bmatrix}
    G_{ac,bc} \\
    G_{ac,cb} \\
    G_{ca,bc} \\
    G_{ca,cb}
\end{bmatrix}.$$ 

A similar equation holds for the block of $Y_{aa,ba}$, $Y_{aa,ab}$, and $Y_{aa,aa}$. The components $G_{ac,bc}$, etc., of the right-hand side are calculated as

$$G_{ac,bc} = E[F_{ac}F_{bc}]E[\varphi(s_a)\varphi(s_b)s_c] = l_{ac}\sigma_c^2$$

eq etc. It is easy to check that all of the components of this type are zero when $l_a = E[\varphi(s_a)] = 0$ for all $a$. This shows that $Y_{ac,bc} = Y_{aa,ba} = 0$. Since $E[\Delta s_a \Delta s_b]$ is given by (7.5) for $a \neq b$ for large $t$

$$E[\Delta s_a \Delta s_b] = O(\eta^2).$$

**VIII. ON-LINE VERSUS BATCH LEARNING**

Finally, we discuss the condition when the true solution is a stable equilibrium of the learning algorithm. We also obtain the best $F$ in the equivalence class. As long as $\eta$ is fixed to a constant, $W_t$ never converges to $W$. It is better to keep $\eta$ finite when the environment is fluctuating. On the other hand, when the learning rate $\eta_t$ depends on $t$, $W_t$ converges to $W$ for large $t$. When $\eta_t$ satisfies certain conditions like

$$\sum \eta_t = \infty, \quad \sum \eta_t^2 < \infty \quad (8.1)$$

$W_t$ is known to converge to $W$ (stochastic approximation). In this case, we can compare the asymptotic behaviors of $W_t$ by on-line learning and by batch learning (statistical estimation).

It is, in general, true that the efficiency of on-line learning is worse than batch learning because each training example can be used only once in on-line learning when observed, whereas it can be used repeatedly in batch learning. However, Amari [3], [4] and Opper [25] have proved that there exists an on-line algorithm that gives, asymptotically, the same efficiency as the optimal batch algorithm. This is true in the regular statistical estimation where the Fisher information exists. (It was proved that the efficiency of on-line learning is one half of batch learning in the case of noiseless binary perceptrons where the Fisher information does not exist [29].

We now extend the above result to the case of blind separation.

**Theorem 5:** The natural gradient’s on-line learning rule by standardized estimating function

$$\delta X_t = \frac{1}{t} F^\nu(x_t) \quad (8.2)$$

or

$$W_{t+1} = W_t - \frac{1}{t} F^\nu(x_t; W_t) W_t \quad (8.3)$$
gives, asymptotically, the best performance, which is the same as the optimal batch estimator. It is superefficient under Condition A. Moreover, the true solution is asymptotically stable.

**Proof:** See Appendix D.

It has already been proved in [8] that the learning rule (8.3) is always locally stable at its true solution.

**IX. CONCLUSIONS**

Statistical and dynamical analysis has been made of the problem of blind separation of independent signals within a framework of estimating functions. To this end, a standardized estimating function was introduced and calculated explicitly, proving that superefficiency exists under certain conditions and explaining why source separation works well.

On-line learning was then analyzed. The optimal on-line learning algorithm was explicitly given in terms of the standardized estimating function and its superefficiency proved in the case of a constant and a time-dependent learning rate.

**APPENDIX A**

**CALCULATION OF $K = (K_{ab,cd})$**

In order to calculate the gradient of $F$ with respect to $X$, we put

$$dF(x, W) = F(x, W + dW) - F(x, W)$$

$$= F(x, W + dWX) - F(x, W)$$

where $dF$ denotes the increment of $F$ due to change $dW$ of $W$ and expand it in the form

$$dF_{ab}(x, W) = \sum M_{ab,cd}(x, W)dX_{cd}.$$ 

We then have

$$M_{ab,cd} = \frac{\partial F_{ab}}{\partial X_{cd}}$$

and its expectation gives $K_{ab,cd}$. For $F = (F_{ab})$ given by (3.6)

$$F_{ab} = \varphi(y_a)y_b - \delta_{ab}$$

we have

$$dF_{ab} = d\varphi(y_a)y_b + \varphi(y_a)dy_b$$

$$\varphi(y_a)dX_{cd} + \varphi(y_a)dy_b.$$ 

From

$$dy = dWy = dWW^{-1}X = dXy$$
we have

\[ d\hat{y}_a = \sum_{d=1}^{n} dX_{ad}dy_t = \sum_{c,d=1}^{n} y_t \delta_{ac} dX_{cd}. \]

Therefore

\[ M_{ab, cd} = \varphi'(y_a) y_b \varphi'(y_c) + \varphi'(y_a) y_b \varphi'(y_c). \]

At the true \( W \), \( y_a \) and \( y_b \) are independent for \( \alpha \neq b \). Hence

\[
\begin{align*}
E[\varphi'(y_a) y_b y_c] &= E[\varphi'(y_a) y_b] K_{ac} \delta_{bd} \\
E[\varphi'(y_a) y_b] &= \delta_{ad}
\end{align*}
\]

because of (2.4) and

\[ h_\alpha(s_a) = \varphi(s_a) s_a - 1. \]

This proves the lemma.

**APPENDIX B**

**Calculation of \( G^*_a, b \)**

Since \( F^* \) is given in (5.10) and (5.11), we can evaluate them easily as

\[
G^*_{ac, bc} = c_{ac} c_{bc} E \left[ \left( k_c \sigma^2 \varphi'(y_a) y_c - \varphi'(y_c) y_a \right) \cdot \left( k_b \sigma^2 \varphi'(y_b) y_c - \varphi'(y_c) y_b \right) \right]
\]

\[ = c_{ac} c_{bc} k_c k_b \sigma^4 \left[ E \left[ \varphi'(y_a) y_c \right] \right] + \left[ E \left[ \varphi'(y_c) y_a \right] \right] \]

\[ = c_{ac} c_{bc} k_c k_b \sigma^4 \left[ E \left[ \varphi'(y_a) y_c \right] \right] + \left[ E \left[ \varphi'(y_c) y_a \right] \right] \]

\[ G^*_{ac, bc} = \frac{1}{m_a + 1} \frac{c_{ac} c_{bc}}{m_a + 1} E \left[ \left( \varphi'(y_c) y_c - 1 \right) \cdot \left( \varphi'(y_c) y_c - 1 \right) \right] \]

\[ \cdot \left[ \sigma^2 \varphi'(y_c) y_c - \varphi'(y_c) y_c \right] \]

\[ = \frac{1}{m_a + 1} \frac{c_{ac} k_c \sigma^2 k_b \sigma^2}{E \left[ \varphi'(y_c) y_c \right]}. \]

**APPENDIX C**

**Derivation of (7.4) for Covariances of Errors by Learning**

Let us use the extended suffix \( A = (a, b) \) for a pair \( (a, b) \) of indices. The learning equation (7.3) can be rewritten in terms of errors

\[ \Delta X_t = (W_t - W) W^{-1} \]

as

\[ \Delta X_{t+1} = \Delta X_t - \eta F(x, W_t). \]

When \( W_t \) is close to \( W \) and \( \Delta X_t \) is small, we have

\[ \Delta X_{t+1} = \Delta X_t - \eta \left\{ F(x, W_t) + \frac{\partial F}{\partial X} \Delta X_t \right\} \]

by the Taylor expansion. Let

\[ Z^*_A = E[\Delta X_A^t \Delta X_A^t] \]

be the error covariances at time \( t \). We then have

\[ Z^{t+1}_A = Z^*_A - \eta E \left[ \Delta X_A^t \sum C_{BC} \Delta X_C^t \right] \]

\[ + \Delta X_B^t \sum K_{AC} C_{AC} \Delta X_C^t \]

\[ + \eta^2 E[F_A F_B] + \text{higher order terms.} \]

Therefore, when \( W_t \) converges to \( W \), for sufficiently large \( t \), we have

\[
\sum_C E[\Delta X_A^t \Delta X_C^t] K_{BC} + E[\Delta X_B^t \Delta X_C^t] K_{AC} = \eta G_{AB} + O(\eta^2)
\]

proving (7.4).

**APPENDIX D**

**Proof of Theorem 5**

The error term \( \Delta X_t \) is given by

\[ \Delta X_{t+1} = \Delta X_t - \frac{1}{t} F^*(x, W_t) \]

(see Appendix B). By expansion, we have

\[ \Delta X_{t+1} = \Delta X_t - \frac{1}{t} \left\{ F^*(x, W_t) + \frac{\partial F^*}{\partial X} \Delta X_t \right\} \]

However, in the present case

\[ K^* = \frac{\partial F^*}{\partial X} \]

is the identity operator. Hence, for

\[ Z_t = E[\Delta X_t \Delta X_t] \]

we have

\[ Z_{t+1} = \left( 1 - \frac{2}{t} \right) Z_t + \frac{1}{t^2} G^* + \text{higher-order terms.} \]

The solution of this equation is

\[ Z_t = \frac{1}{t} G^* + O \left( \frac{1}{t^2} \right) \]

which is asymptotically the same as the optimal batch solution (6.3).

**REFERENCES**


Shun-ichi Amari (F’88) was born in Tokyo, Japan, on January 3, 1936. He graduated from the University of Tokyo in 1958, having majored in mathematical engineering and received the Dr.Eng. degree from the University of Tokyo in 1963.

He was an Associate Professor at Kyushu University, Kyushu, Japan, an Associate and then Full Professor at the Department of Mathematical Engineering and Information Physics, University of Tokyo, and is now Professor Emeritus at the University of Tokyo. He is the Director of the Brain-Style Information Systems Group, RIKEN Brain Science Institute, Saitama, Japan. He has been engaged in research in wide areas of mathematical engineering and applied mathematics such as topological network theory, differential geometry of continuum mechanics, pattern recognition, mathematical foundations of neural networks, and information geometry.

Dr. Amari served as President of the International Neural Network Society, Council Member of the Bernoulli Society for Mathematical Statistics and Probability Theory, and Vice President of the Institute of Electrical, Information and Communication Engineers. He was founding Coeditor-in-Chief of Neural Networks. He has been awarded the Japan Academy Award, the IEEE Neural Networks Pioneer Award, and the IEEE Emanuel R. Piore Award.